



**Universidad Autónoma del Estado de Hidalgo**

**Instituto de Ciencias Básicas e Ingeniería**

**Licenciatura en Ciencias Computacionales**

# **Proyecto Base de datos distribuidas para una biblioteca**

**Sexto semestre, Grupo dos**

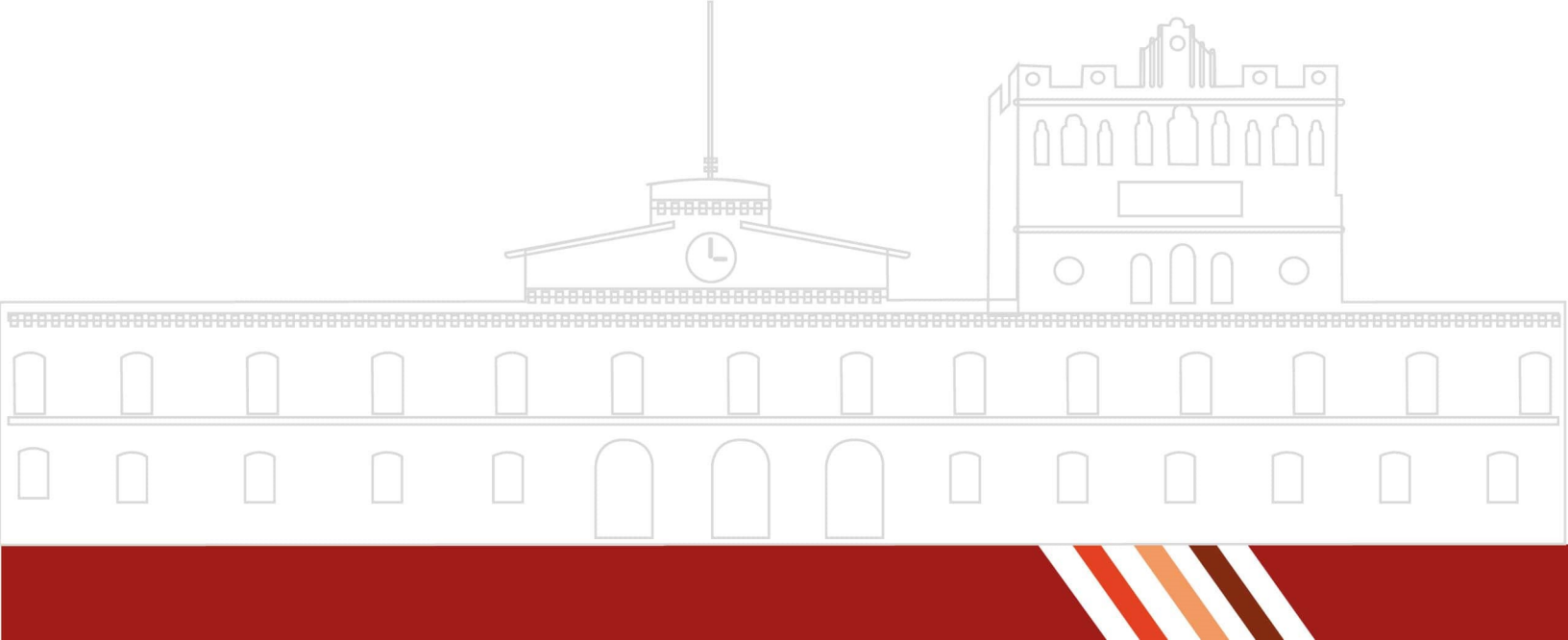
**Catedrático: Dr. Eduardo Cornejo Velzquez**

**Base de datos distribuidas**

Equipo:

Juan Carlos Montes Gonzalez

Jennifer Resendiz Isidro



# Contents

<b>1</b>	<b>Introducción</b>	<b>3</b>
<b>2</b>	<b>Marco teórico</b>	<b>3</b>
2.1	Requerimientos . . . . .	4
2.2	Arquitectura general . . . . .	5
<b>3</b>	<b>Desarrollo</b>	<b>5</b>
3.1	Modelo Entidad - Relación . . . . .	6
<b>4</b>	<b>SQL</b>	<b>6</b>
4.1	Creación de tablas . . . . .	6
<b>5</b>	<b>Fragmentación</b>	<b>7</b>
<b>6</b>	<b>Desarrollo</b>	<b>9</b>
<b>7</b>	<b>Interfaces</b>	<b>9</b>
7.1	Biblioteca ICEA . . . . .	10
7.2	Biblioteca Central . . . . .	13
<b>8</b>	<b>Herramientas empleadas</b>	<b>14</b>
<b>9</b>	<b>Conclusiones</b>	<b>14</b>
<b>10</b>	<b>Referencias Bibliográficas</b>	<b>14</b>

## Introducción

El siguiente proyecto busca diseñar e implementar un sistema de base de datos para una biblioteca universitaria la cual tiene diferentes ubicaciones para cada instituto con sus respectivos libros. El sistema se usará para el manejo de datos, mejorando la eficiencia de los usuarios y clientes de las bibliotecas. Se enfocará en el uso para el administrador, quien podrá manejar los datos de quien solicita el libro, administrar los préstamos como consultas de las fechas en la que se solicita el libro, la fecha que se espera la devolución, penalización por entregas atrasadas, nombre de quien lo solicita y las reservaciones. Permitiendo consultas de información sin ambigüedad al igual que actualizar estos mismos datos de manera simultánea.

Contar con un catálogo que deberá poder actualizarse desde cualquier nodo de forma automática, clasificados por nombre, autor y categoría, teniendo en cuenta que puede existir más de una copia de un mismo libro o diferentes editoriales. Mostrando la ubicación de dicho libro y su disponibilidad dentro de las bibliotecas.

La distribución tendrá fragmentación horizontal por ubicación, se busca que cada nodo opere de manera independiente y a su vez tendremos una vista global que integre a ambos. Tablas como usuarios y/o administradores tendrán réplicas en ambos nodos para mantener consistencia.

Aunque nos enfocaremos más en la eficiencia de la base de datos, el sistema tendrá una capa de interfaz, fácil de leer tomando en cuenta que el administrador recibirá entrenamiento para usarlo.

## Marco teórico

**Base de datos distribuidas** Una base de datos distribuida es un sistema de gestión de bases de datos en el que las bases de datos no están ubicadas en un sistema de servidor común. Las bases de datos pueden almacenarse en varios servidores de datos ubicados en la misma ubicación física, o pueden distribuirse a través de una red de área local o amplia. Las actualizaciones a una base de datos distribuida pueden llevarse a cabo contra una sola base de datos como una unidad de trabajo remota o a través de varias bases de datos como una actualización multisitio.

**Fragmentación de Bases de Datos** La fragmentación es una técnica de optimización que distribuye las tablas entre otros servidores de bases de datos. Es como el particionamiento en el sentido de que ambos implican la división de los datos en subconjuntos más pequeños. La diferencia es que la fragmentación distribuye estos subconjuntos a diferentes servidores, mientras que el particionamiento los almacena en una sola base de datos. Estos servidores utilizan el mismo motor de base de datos y el mismo tipo de hardware para conseguir un nivel de rendimiento similar para todos los fragmentos.

El objetivo de la fragmentación es lograr una arquitectura de no compartir nada, eliminando los cuellos de botella de procesamiento y los puntos críticos de fallo.

**Fragmentación horizontal** El escalado horizontal es más flexible. Te permite escalar tu base de datos según sea necesario, añadiendo más servidores a tu sistema. Cada uno de estos servidores proporciona recursos a diferentes fragmentos de la base de datos. Esto distribuye la carga de trabajo y mejora la capacidad del sistema para gestionar más peticiones.

## Requerimientos

### Book (libro)

- isbn *primary key*
- title *varchar*
- Location *varchar*
- book\_status *varchar*

### Author (autor)

- id\_author *primary key*
- name *varchar*

### Category (categoria)

- id\_category *primary key*
- category\_name *varchar*

### Publisher (editorial)

- id\_publisher *primary key*
- publisher\_name *varchar*

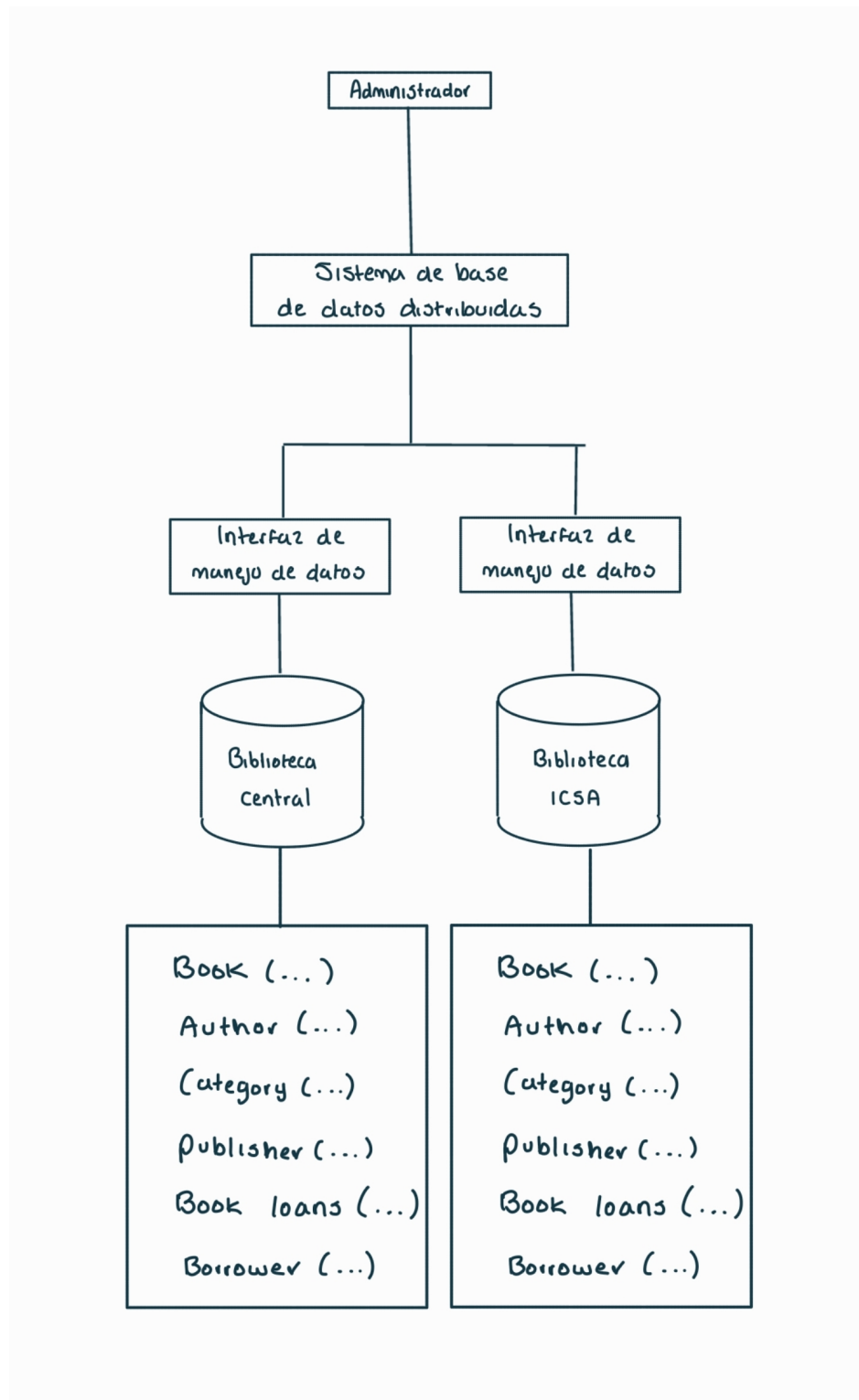
### Book Loans (prstamos de libros)

- id\_loan *primary key*
- start\_date *timestamp*
- end\_date *timestamp*
- return\_date *timestamp*
- penalty *decimal*
- book\_status *varchar*

### Borrower (prestatario)

- id\_borrower *varchar*
- first\_name *varchar*
- last\_name *varchar*
- second\_lastname *varchar*
- email *varchar*
- phone\_number *varchar*
- status *varchar*

## Arquitectura general



## Desarrollo

## Modelo Entidad - Relacin

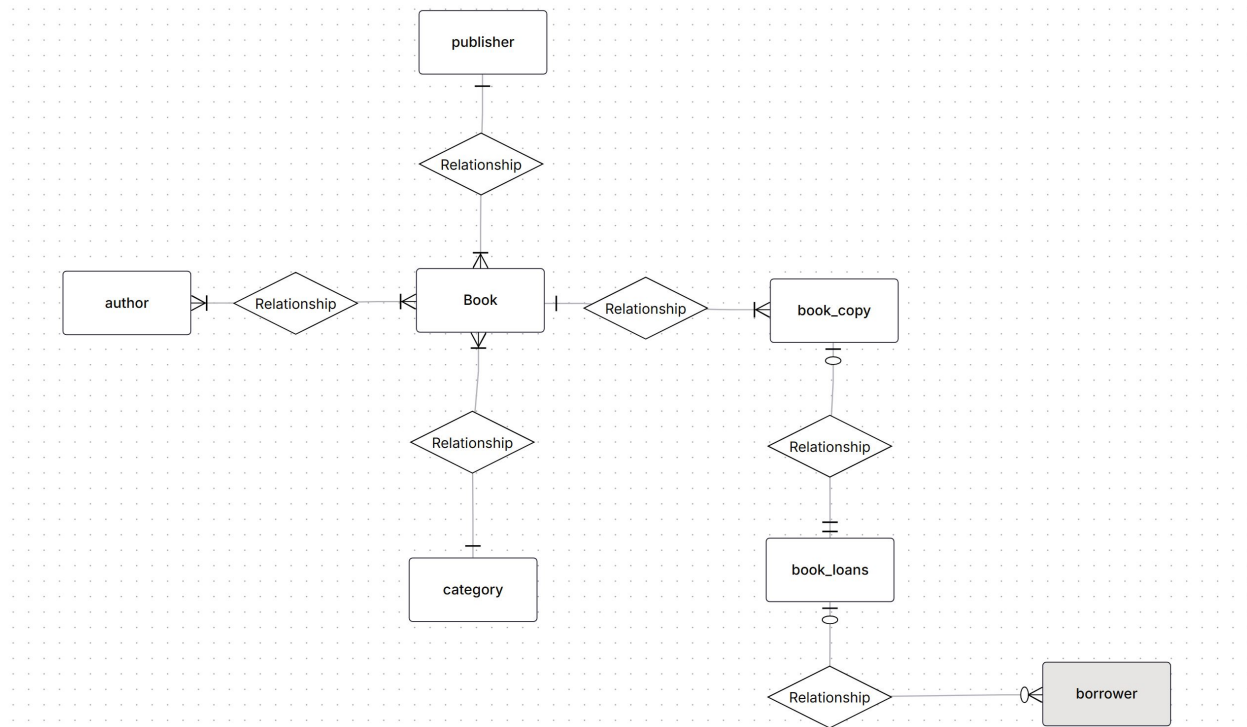


Figure 1: Enter Caption

## SQL

### Creacin de tablas

Listing 1: Creacin de tablas de la base de datos

```
CREATE TABLE Publisher (  
    id_publisher    VARCHAR(50) PRIMARY KEY,  
    publisher_name  VARCHAR(255) NOT NULL  
);  
  
CREATE TABLE Category (  
    id_category     VARCHAR(50) PRIMARY KEY,  
    category_name   VARCHAR(255) NOT NULL  
);  
  
CREATE TABLE Author (  
    id_author       VARCHAR(50) PRIMARY KEY,  
    name            VARCHAR(255) NOT NULL  
);  
  
CREATE TABLE Book (  
    isbn            VARCHAR(50) PRIMARY KEY,  
    title           VARCHAR(255) NOT NULL,  
    book_status     VARCHAR(50),  
    location        VARCHAR(255),  
    id_publisher    VARCHAR(50),
```

```

        id_category          VARCHAR(50),
        FOREIGN KEY (id_publisher) REFERENCES Publisher(id_publisher),
        FOREIGN KEY (id_category) REFERENCES Category(id_category)
    );

CREATE TABLE Book_Author (
    isbn          VARCHAR(50),
    id_author     VARCHAR(50),
    PRIMARY KEY (isbn, id_author),
    FOREIGN KEY (isbn) REFERENCES Book(isbn),
    FOREIGN KEY (id_author) REFERENCES Author(id_author)
);

CREATE TABLE Borrower (
    id_borrower   VARCHAR(50) PRIMARY KEY,
    first_name    VARCHAR(255) NOT NULL,
    last_name     VARCHAR(255) NOT NULL,
    second_lastname VARCHAR(255),
    email         VARCHAR(255),
    phone_number  VARCHAR(20),
    status        VARCHAR(50)
);

CREATE TABLE Book_Loans (
    id_loan       VARCHAR(50) PRIMARY KEY,
    start_date    TIMESTAMP NOT NULL,
    end_date      TIMESTAMP NOT NULL,
    return_date   TIMESTAMP,
    penalty       DECIMAL(10,2),
    book_status   VARCHAR(50),
    isbn         VARCHAR(50),
    id_borrower   VARCHAR(50),
    FOREIGN KEY (isbn) REFERENCES Book(isbn),
    FOREIGN KEY (id_borrower) REFERENCES Borrower(id_borrower)
);

```

## Fragmentacin

Listing 2: Sentencias SQL para fragmentacin horizontal de la biblioteca

```

CREATE DATABASE biblioteca_icea;
USE biblioteca_icea;

-- Tablas generales
CREATE TABLE Publisher (
    id_publisher   VARCHAR(50) PRIMARY KEY,
    publisher_name VARCHAR(255) NOT NULL
);

CREATE TABLE Category (
    id_category   VARCHAR(50) PRIMARY KEY,
    category_name VARCHAR(255) NOT NULL
);

CREATE TABLE Author (
    id_author     VARCHAR(50) PRIMARY KEY,
    name          VARCHAR(255) NOT NULL
);

```

```

);

CREATE TABLE Borrower (
    id_borrower    VARCHAR(50) PRIMARY KEY,
    first_name     VARCHAR(255) NOT NULL,
    last_name      VARCHAR(255) NOT NULL,
    second_lastname VARCHAR(255),
    email          VARCHAR(255),
    phone_number   VARCHAR(20),
    status         VARCHAR(50)
);

-- Fragmento ICEA
CREATE TABLE Book_ICEA (
    isbn           VARCHAR(50) PRIMARY KEY,
    title          VARCHAR(255) NOT NULL,
    book_status    VARCHAR(50),
    location       VARCHAR(255),
    id_publisher   VARCHAR(50),
    id_category    VARCHAR(50),
    CHECK (location = 'ICEA'),
    FOREIGN KEY (id_publisher) REFERENCES Publisher(id_publisher),
    FOREIGN KEY (id_category) REFERENCES Category(id_category)
);

-- Fragmento CENTRAL
CREATE TABLE Book_CENTRAL (
    isbn           VARCHAR(50) PRIMARY KEY,
    title          VARCHAR(255) NOT NULL,
    book_status    VARCHAR(50),
    location       VARCHAR(255),
    id_publisher   VARCHAR(50),
    id_category    VARCHAR(50),
    CHECK (location = 'CENTRAL'),
    FOREIGN KEY (id_publisher) REFERENCES Publisher(id_publisher),
    FOREIGN KEY (id_category) REFERENCES Category(id_category)
);

-- Relaciones Autor-Libro en ICEA
CREATE TABLE Book_Author_ICEA (
    isbn           VARCHAR(50),
    id_author      VARCHAR(50),
    PRIMARY KEY (isbn, id_author),
    FOREIGN KEY (isbn) REFERENCES Book_ICEA(isbn),
    FOREIGN KEY (id_author) REFERENCES Author(id_author)
);

-- Relaciones Autor-Libro en CENTRAL
CREATE TABLE Book_Author_CENTRAL (
    isbn           VARCHAR(50),
    id_author      VARCHAR(50),
    PRIMARY KEY (isbn, id_author),
    FOREIGN KEY (isbn) REFERENCES Book_CENTRAL(isbn),
    FOREIGN KEY (id_author) REFERENCES Author(id_author)
);

-- Pr stamos en ICEA
CREATE TABLE Book_Loans_ICEA (
    id_loan        VARCHAR(50) PRIMARY KEY,

```



```

start_date      VARCHAR(60) NOT NULL,
end_date        VARCHAR(60) NOT NULL,
return_date     VARCHAR(60),
penalty         DECIMAL(10,2),
book_status     VARCHAR(50),
isbn            VARCHAR(50),
id_borrower     VARCHAR(50),
CHECK (isbn IN (SELECT isbn FROM Book_ICEA)),
FOREIGN KEY (isbn) REFERENCES Book_ICEA(isbn),
FOREIGN KEY (id_borrower) REFERENCES Borrower(id_borrower)
);

-- Pr stamos en CENTRAL
CREATE TABLE Book_Loans_CENTRAL (
  id_loan        VARCHAR(50) PRIMARY KEY,
  start_date     VARCHAR(60) NOT NULL,
  end_date       VARCHAR(60) NOT NULL,
  return_date    VARCHAR(60),
  penalty        DECIMAL(10,2),
  book_status    VARCHAR(50),
  isbn           VARCHAR(50),
  id_borrower    VARCHAR(50),
  CHECK (isbn IN (SELECT isbn FROM Book_CENTRAL)),
  FOREIGN KEY (isbn) REFERENCES Book_CENTRAL(isbn),
  FOREIGN KEY (id_borrower) REFERENCES Borrower(id_borrower)
);

```

## Desarrollo

En los equipos la conexin remoto debe estar habilitado al igual que el firewall y el antivirus de las computadoras que usamos, generamos un una direccin IP que sera el hostname (la cual cambiaba de direccin dependiendo la conexin a internet que tenamos).

En el equipo remoto (ICEA) obtuvimos una ID/Usuario y una contrasea, para luego usarlo en los equipos y el codigo en visual studio code.

En la IDE de visual studio code, realizamos interfaces con Python y descargamos la libreria de import mysql.connector la cual permiti conectar la base a la interfaz.

## Interfaces

Hechos con Python

# Biblioteca ICEA

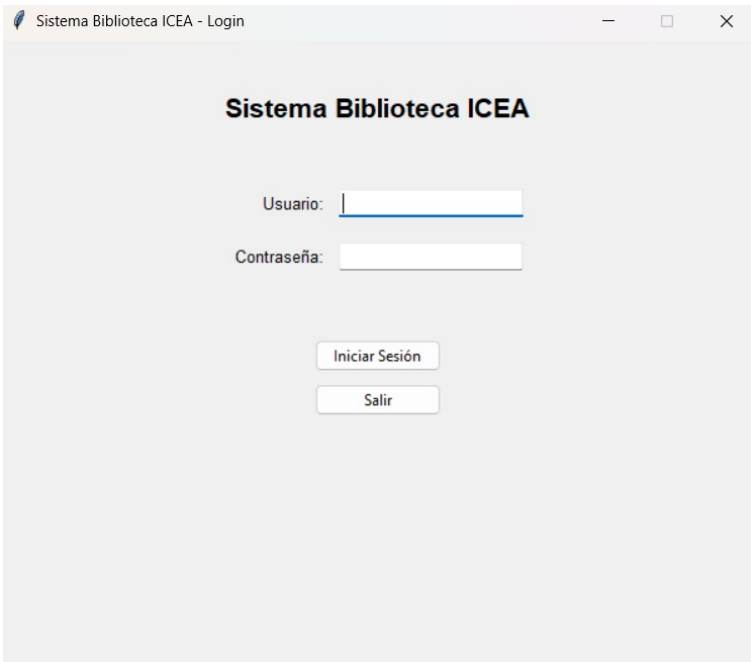


Figure 2: Inicio

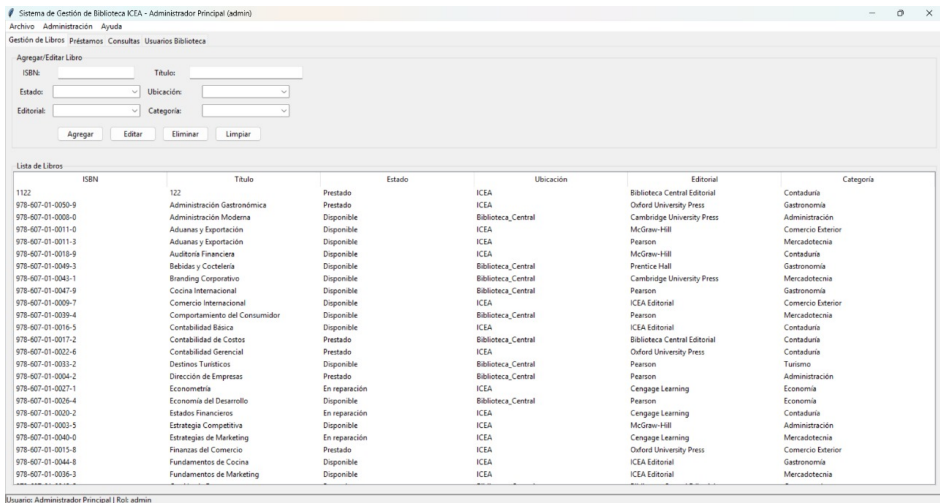


Figure 3: Gestion de libros

Sistema de Gestión de Biblioteca ICEA - Administrador Principal (admin)

Archivo Administración Ayuda

Gestión de Libros Préstamos Consultas Usuarios Biblioteca

Gestionar Préstamo

ID Préstamo:  Libro (ISBN):

Usuario:  Fecha Inicio: 2025-11-13 18:26:41

Fecha Fin:  Estado:

ID	Libro	Usuario	Inicio	Fin	Estado
LOAN050	Administración Gastronomía	Mauricio Palacios	2024-04-24 11:05:00	2024-05-08 11:05:00	Prestado
LOAN047	Cocina Internacional	Silvia Márquez	2024-04-18 15:40:00	2024-05-02 15:40:00	Prestado
LOAN045	Gestión de Restaurantes	Valeria Domínguez	2024-04-14 13:20:00	2024-04-28 13:20:00	Prestado
LOAN042	Marketing de Servicios	Rodrigo Soto	2024-04-08 09:30:00	2024-04-22 09:30:00	Prestado
LOAN040	Estrategias de Marketing	Andrés Pérez	2024-04-04 15:35:00	2024-04-18 15:35:00	Prestado
LOAN037	Marketing Digital	Karla Delgado	2024-03-29 12:05:00	2024-04-12 12:05:00	Prestado
LOAN035	Planificación Turística	Daniela Navarro	2024-03-25 09:45:00	2024-04-08 09:45:00	Prestado
LOAN032	Marketing Turístico	Manuel Santos	2024-03-19 14:20:00	2024-04-02 14:20:00	Prestado
LOAN030	Gestión Turística	Francisco Vázquez	2024-03-15 11:55:00	2024-03-29 11:55:00	Prestado
LOAN027	Econometría	Claudia Moreno	2024-03-09 16:30:00	2024-03-23 16:30:00	Prestado
LOAN025	Macroeconomía	Lucía Juárez	2024-03-05 14:15:00	2024-03-19 14:15:00	Prestado
LOAN022	Contabilidad General	Sergio Aguilar	2024-02-28 10:40:00	2024-03-13 10:40:00	Prestado
LOAN020	Estadística Financiera	Arturo Luna	2024-02-24 16:40:00	2024-03-09 16:40:00	Prestado
LOAN017	Contabilidad de Costos	Carmen Salazar	2024-02-18 13:15:00	2024-03-03 13:15:00	Prestado
LOAN015	Finanzas del Comercio	Gabriela Guerrero	2024-02-14 10:30:00	2024-02-28 10:30:00	Prestado
LOAN012	Logística Internacional	Ricardo Ortiz	2024-02-08 15:30:00	2024-02-22 15:30:00	Prestado
LOAN010	Tratados Comerciales	Diego Reyes	2024-02-03 13:20:00	2024-02-17 13:20:00	Prestado
LOAN007	Gestión del Cambio	Isabel Ramírez	2024-01-25 09:30:00	2024-02-08 09:30:00	Prestado
LOAN005	Teoría Organizacional	Laura González	2024-01-20 15:20:00	2024-02-03 15:20:00	Prestado
LOAN003	Estrategia Competitiva	Ana Hernández	2024-01-15 11:20:00	2024-01-29 11:20:00	Prestado

Usuario: Administrador Principal | Rol: admin

Figure 4: prestamos

Sistema de Gestión de Biblioteca ICEA - Administrador Principal (admin)

Archivo Administración Ayuda

Gestión de Libros Préstamos Consultas Usuarios Biblioteca

ISBN	Título	Ubicación
123456	Juan Carlos	ICEA
978-007-01-0001-1	Principios de Administración	ICEA
978-007-01-0003-5	Estrategia Competitiva	ICEA
978-007-01-0006-6	Liderazgo Efectivo	Biblioteca_Central
978-007-01-0008-0	Administración Moderna	Biblioteca_Central
978-007-01-0009-7	Comercio Internacional	ICEA
978-007-01-0011-0	Aduanas y Exportación	ICEA
978-007-01-0011-3	Aduanas y Exportación	ICEA
978-007-01-0012-7	Logística Internacional	Biblioteca_Central
978-007-01-0014-1	Marketing Internacional	Biblioteca_Central
978-007-01-0016-5	Contabilidad Básica	ICEA
978-007-01-0018-9	Auditoría Financiera	ICEA
978-007-01-0019-6	Impuestos Empresariales	Biblioteca_Central
978-007-01-0021-9	Normas Contables	Biblioteca_Central
978-007-01-0023-3	Principios de Economía	ICEA
978-007-01-0025-7	Macroeconomía	ICEA
978-007-01-0026-4	Economía del Desarrollo	Biblioteca_Central
978-007-01-0028-8	Historia Económica	Biblioteca_Central
978-007-01-0030-1	Gestión Turística	ICEA
978-007-01-0032-5	Marketing Turístico	ICEA
978-007-01-0033-2	Destinos Turísticos	Biblioteca_Central
978-007-01-0035-6	Planificación Turística	Biblioteca_Central
978-007-01-0036-5	Fundamentos de Marketing	ICEA
978-007-01-0038-7	Investigación de Mercados	ICEA
978-007-01-0039-4	Comportamiento del Consumidor	Biblioteca_Central
978-007-01-0041-7	Publicidad y Promoción	Biblioteca_Central

Usuario: Administrador Principal | Rol: admin

Figure 5: Consultas

Sistema de Gestión de Biblioteca ICA - Administrador Principal (admin)

Archivo Administración Ayuda

Gestión de Libros Préstamos Consultas Usuarios Biblioteca

Libros Disponibles

Préstamos Activos

Libros por Categoría

Usuarios con Multas

Categoría	Total Libros
Marketing	10
Administración	8
Contabilidad	8
Comercio Exterior	7
Economía	7
Gastronomía	7
Turismo	6
Finanzas	0
Recursos Humanos	0
Emprendimiento	0

Usuario: Administrador Principal | Rol: admin

Figure 7: Libros por categoria

Sistema de Gestión de Biblioteca ICA - Administrador Principal (admin)

Archivo Administración Ayuda

Gestión de Libros Préstamos Consultas Usuarios Biblioteca

Libros Disponibles

Préstamos Activos

Libros por Categoría

Usuarios con Multas

Nombre	Apellido	Email	Total Multas
Carlos	Rodriguez	carlos.rodriguez@email.com	50.00

Usuario: Administrador Principal | Rol: admin

Figure 6: Usuarios con multas

Sistema de Gestión de Biblioteca ICA - Administrador Principal (admin)

Archivo Administración Ayuda

Gestión de Libros Préstamos Consultas Usuarios Biblioteca

Agregar/Editar Usuario Biblioteca

ID Usuario:  Nombre:

Apellido Paterno:  Apellido Materno:

Email:  Teléfono:

Estado:

Lista de Usuarios Biblioteca

ID	Nombre	Apellidos	Email	Teléfono	Estado
BOR001	Maria	García López	maria.garcia@email.com	555-1001	Activo
BOR002	Carlos	Rodríguez Martínez	carlos.rodriguez@email.com	555-1002	Activo
BOR003	Ana	Hernández Gómez	ana.hernandez@email.com	555-1003	Activo
BOR004	Luis	Martínez Díaz	luis.martinez@email.com	555-1004	Inactivo
BOR005	Laura	González Pérez	laura.gonzalez@email.com	555-1005	Activo
BOR006	Javier	Sánchez Romero	javier.sanchez@email.com	555-1006	Activo
BOR007	Isabel	Ramírez Torres	isabel.ramirez@email.com	555-1007	Activo
BOR008	Miguel	Flores Castro	miguel.flores@email.com	555-1008	Inactivo
BOR009	Elena	Montero Ortega	elena.montero@email.com	555-1009	Activo
BOR010	Diego	Reyes Silva	diego.reyes@email.com	555-1010	Activo
BOR011	Patricia	Cruz Mendoza	patricia.cruz@email.com	555-1011	Activo
BOR012	Ricardo	Ortiz Ríos	ricardo.ortiz@email.com	555-1012	Activo
BOR013	Sofía	Chávez Vega	sofia.chavez@email.com	555-1013	Inactivo
BOR014	Fernando	Medina Campos	fernando.medina@email.com	555-1014	Activo
BOR015	Gabriela	Guerrero Rojas	gabriela.guerrero@email.com	555-1015	Activo
BOR016	Roberto	Paredes Miranda	roberto.paredes@email.com	555-1016	Activo
BOR017	Carmen	Salazar Núñez	carmen.salazar@email.com	555-1017	Activo
BOR018	Oscar	Vargas Mejía	oscar.vargas@email.com	555-1018	Inactivo
BOR019	Adriana	Cortés León	adriana.cortes@email.com	555-1019	Activo
BOR020	Arturo	Luna Espinoza	arturo.luna@email.com	555-1020	Activo
BOR021	Rosa	Méndez Fuentes	rosa.mendez@email.com	555-1021	Activo
BOR022	Sergio	Aguilar Valdez	sergio.aguilar@email.com	555-1022	Activo
BOR023	Teresa	Castillo Ochoa	teresa.castillo@email.com	555-1023	Inactivo

Usuario: Administrador Principal | Rol: admin

Figure 8: Usuarios de biblioteca

## Biblioteca Central

Sistema Biblioteca Central - Login

**Sistema Biblioteca Central**

Usuario:

Contraseña:

Figure 9: Inicio Biblioteca central

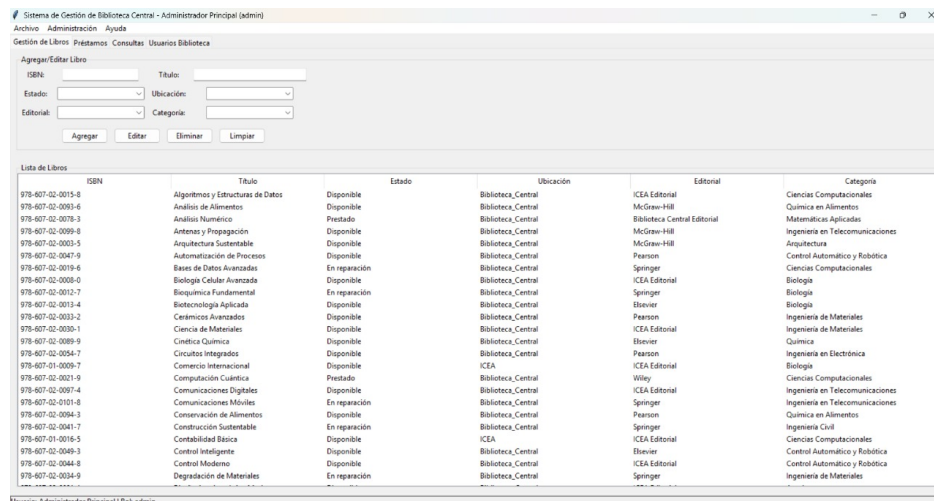


Figure 10: Gestión de libros Biblioteca Central

## Herramientas empleadas

**MySQL Workbench** MySQL Workbench es una herramienta visual unificada para arquitectos, desarrolladores y administradores de bases de datos. Ofrece modelado de datos, desarrollo SQL y herramientas de administración completas para la configuración del servidor, la administración de usuarios, las copias de seguridad y mucho más. Está disponible para Windows, Linux y macOS.

**Python Tinker** El paquete tkinter (interfaz Tk) es la interfaz por defecto de Python para el kit de herramientas de GUI Tk. Tanto Tk como tkinter están disponibles en la mayoría de las plataformas Unix, así como en sistemas Windows (Tk en sí no es parte de Python, es mantenido por ActiveState).

**Visual Studio Code** Es un editor de código para programadores gratuito, de código abierto y multiplataforma. Está desarrollado por Microsoft, una compañía con una dilatada experiencia en la creación de IDEs (entornos de desarrollo integrados), que ha conseguido plasmar su larga tradición en el sector para ofrecer una herramienta ligera y práctica que la comunidad ha adoptado en masa.

## Conclusiones

Este proyecto nos permitió aplicar métodos y estrategias aprendidas durante el curso, haciendo una base de datos distribuida en base a una problemática real. Nos cuenta que es más difícil de lo que pensamos y teníamos anticipado, y no solo se deben aplicar conocimientos del curso de Base de datos distribuidas, sino que debemos saber aplicar conocimiento de todas las demás.

## Referencias Bibliográficas

### References

- [1] Arsys. (s. f.). *¿Qué es Visual Studio Code y cuáles son sus ventajas?* <https://www.arsys.es/blog/que-es-visual-studio-code-y-cuales-son-sus-ventajas>
- [2] IBM. (s. f.). *Db2 for Linux, UNIX and Windows*. <https://www.ibm.com/docs/es/db2/11.1.0?topic=managers-designing-distributed-databases>
- [3] MySQL. (s. f.). *MySQL Workbench*. <https://www.mysql.com/products/workbench/>

- [4] Ravoof, S. (2023, agosto 23). *Una gua para entender la fragmentacin de las bases de datos*. Kinsta.  
<https://kinsta.com/es/blog/fragmentacion-de-la-base-de-datos/>
- [5] Python Software Foundation. (s. f.). *tkinter Python interface to Tcl/Tk*. Python Documentation.  
<https://docs.python.org/es/3/library/tkinter.html>